Escape game Q learning

Professor : Frédéric Giroire

Done by :

- Aqabli Souad
- Doubali Salma
- Giuseppe Spathis
- Mesly Houda

1. Initialize the Q-table

> There are 6 states (0,1,2,3,4,5) and 6 actions (go to 0…5).
> Initialize all entries to 0 TD-Q-Learning
>
> $$Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

2. First episode : Look at the second row (state 1) of matrix R. There are two possible actions for the current state 1: go to state 3, or go to state 5. By random selection, we select to go to 5 as our action.

> - 2.a Using Bellman equation, compute the new value of Q(1,5).
>
> > From the reward matrix R: $R(1,5) = 100$. The next state is $s' = 5$.
> >
> > Since state 5 is the goal/terminal state, $\max_{a'} Q(5,a') = 0$.
> >
> > $$Q(1,5) = R(1,5) + \gamma \max_{a'} Q(5,a') = 100 + 0.8 \cdot 0 = 100$$
> >
> > So, $Q(1,5) = 100$.
>
> - 2.b Update the Q-table accordingly.
>
> > Only one entry changes (row 1, col 5):
> >
> > $$Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 100 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$
> >
> > Then the agent reaches state 5 (goal), so the episode ends.

The next state, 5, now becomes the current state. Because 5 is the goal state, we've finished one episode. Our agent's brain now contains an updated matrix Q.

3. Second episode : For the next episode, we start with a randomly chosen initial state. This time, we have state 3 as our initial state and by random selection, we select to go to state 1 as our action. Update the Q(3,1) value.

   From R: $R(3,1) = 0$. Next state is $s' = 1$.

   Now compute $\max_{a'} Q(1,a')$. From the current Q-table, the best value in row 1 is:

   - $$Q(1,5) = 100$$
   - (others are 0)

     So $\max_{a'} Q(1,a') = 100$.

   $$Q(3,1) = R(3,1) + \gamma \max_{a'} Q(1,a') = 0 + 0.8 \cdot 100 = 80$$

   So, $Q(3,1) = 80$.

   The next state, 1, now becomes the current state. We repeat the inner loop of the Q learning algorithm because state 1 is not the goal state. Assume we select again randomly state 5. What is the new updated Q-table at the end of this episode ?

   From state 1 choose action 5 again:

   $$Q(1,5) = 100 + 0.8 \cdot \max_{a'} Q(5,a') = 100 + 0.8 \cdot 0 = 100$$

   So $Q(1,5)$ stays 100, and the episode ends at the goal state 5.

   Final Q after episode 2:

   $$Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 100 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 80 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

4. Try to explore more episodes to find the Q-table reaching the convergence values.

   A compact way to reach the converged values is to make sure every valid transition gets updated after its “next-state max” is already learned. Below is one valid outcome (the converged Q for γ = 0.8 with terminal state 5 having row 0).

   Converged Q-table (only meaningful/allowed actions shown; others stay 0)

   - From state 0: only action to 4

     $Q(0,4) = 0 + 0.8 \cdot \max Q(4,\cdot) = 0.8 \cdot 100 = 80$
   - From state 1: actions to 3 or 5

     $Q(1,5) = 100$

     $Q(1,3) = 0 + 0.8 \cdot \max Q(3,\cdot) = 0.8 \cdot 80 = 64$

- From state 2: only action to 3

  $Q(2,3) = 0 + 0.8 \cdot 80 = 64$

- From state 3: actions to 1,2,4

  $Q(3,1) = 0 + 0.8 \cdot 100 = 80$
  $Q(3,4) = 0 + 0.8 \cdot 100 = 80$
  $Q(3,2) = 0 + 0.8 \cdot \max Q(2,\cdot) = 0.8 \cdot 64 = 51.2$

- From state 4: actions to 0,3,5

  $Q(4,5) = 100$
  $Q(4,0) = 0 + 0.8 \cdot \max Q(0,\cdot) = 0.8 \cdot 80 = 64$
  $Q(4,3) = 0 + 0.8 \cdot 80 = 64$

- From state 5 (goal/terminal): row is 0 by convention

Full 6×6 matrix layout (rows = state, columns = action/next state)

Columns: 0 1 2 3 4 5

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 80 & 0 \\ 0 & 0 & 0 & 64 & 0 & 100 \\ 0 & 0 & 0 & 64 & 0 & 0 \\ 0 & 80 & 51.2 & 0 & 80 & 0 \\ 64 & 0 & 0 & 64 & 0 & 100 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

5. Once the matrix Q gets close enough to a state of convergence, we know our agent has learned the most optimal paths to the goal state.

   Tracing the best sequences of states is as simple as following the links with the highest values at each state. What is the best sequence to escape the building from room 2 ?

   Use the rule “from the current state, take the action with the highest Q value”

   - From state 2, the only possible move is to 3 (and $Q(2,3) = 64$).
   - From state 3, the best Q values are tied:

     $Q(3,1) = 80$ and $Q(3,4) = 80$.

   So there are two optimal escape sequences (same optimal value):

   1. **2 → 3 → 1 → 5**
   2. **2 → 3 → 4 → 5**

   Both are optimal because at state 3 you can choose either 1 or 4 with equal highest Q.